



## Image segmentaion and restoration using parametric contours with free endpoints

Heike Benninghoff and Harald Garcke

Preprint Nr. 07/2015



## Image segmentaion and restoration using parametric contours with free endpoints

Heike Benninghoff and Harald Garcke

Preprint Nr. 07/2015

# Image Segmentation and Restoration Using Parametric Contours With Free Endpoints

Heike Benninghoff, and Harald Garcke

**Abstract**—In this paper, we introduce a novel approach for active contours with free endpoints. A scheme is presented for image segmentation and restoration based on a discrete version of the Mumford-Shah functional where the contours can be both closed and open curves. Additional to a flow of the curves in normal direction, evolution laws for the tangential flow of the endpoints are derived. Using a parametric approach to describe the evolving contours together with an edge-preserving denoising, we obtain a fast method for image segmentation and restoration. The analytical and numerical schemes are presented followed by numerical experiments with artificial test images and with a real medical image.

**Index Terms**—Image segmentation, image restoration, active contours, Mumford-Shah, Chan-Vese, parametric method, variational methods, free endpoints, open boundaries.

## I. INTRODUCTION

THIS article addresses important classical problems in image processing: image segmentation, edge detection and image restoration.

Image segmentation aims at partitioning a given image into its constituent parts, also called regions or phases. A segmentation of an image can be given by a set of region boundaries and edges. Different types of edges can occur in images: edges can be boundaries of objects and separate these objects from their background or from each other. But edges can also end inside the image at a location where no other edge continues.

Boundaries of objects can be modeled with so-called interface curves. Non-interface curves are curves which do not separate two different regions in the image. Such curves have one or two so-called free endpoints.

Image restoration aims at reducing or removing noise which affects a given image. Typically, a blurring of the sharp edges in the image should be prevented when smoothing an image. This results in the need of an edge preserving image denoising method.

Image segmentation including edge detection can be performed with active contours (also called snakes), first proposed by Kass, Witkin, and Terzopoulos [1] in 1988. Since this time, the popular method is applied and further developed by many authors, e.g. [2]–[10]. Using active contours, a curve evolves in order to minimize a given energy functional. The energy functional should be designed such that a minimizing curve matches with the region boundaries or edges in the image.

The Mumford-Shah functional [11] can be used for both image segmentation and image restoration. A pair  $(\Gamma, u)$  should be found which minimizes the Mumford-Shah energy, where  $\Gamma$  is a set of curves and  $u$  is a piecewise smooth function with possible discontinuities across  $\Gamma$ . Having found a solution  $(\Gamma, u)$ , a segmentation of the image is given by the set of object boundaries and edges  $\Gamma$ , and a denoised version of the image is given by the piecewise smooth approximation  $u$ .

An important variant of the Mumford-Shah problem is the restriction to piecewise *constant* image approximations  $u$ , the so-called minimal partition problem [10]. However if edges with free endpoints, also called crack-tips [11], occur, the piecewise constant approximation will not be applicable.

It is also possible to approximate the Mumford-Shah functional by a sequence of simpler elliptic variational problems as introduced by Ambrosio and Tortorelli [12]. They replaced the curve  $\Gamma$  by a 2D function for which a phase field type energy is added to the functional.

Image segmentation and restoration are classical areas in image analysis, see [1], [3], [11], [13], but still significant in more present research, see e.g. [10], [14]–[22] to mention some selected works. There is also a variety of related image processing tasks like object detection [8], [23] or pattern recognition [24]), feature extraction [25] and anomaly detection [26].

The image segmentation method, considered and developed in this article, also uses the evolution of curves. The resulting evolution equations, derived from the Mumford-Shah functional, can be written as parabolic partial differential equations for a parametrization of the curves  $\Gamma$ . The restoration is performed by solving a diffusion equation for  $u$ , also derived from the Mumford-Shah model. By using the location of the curves  $\Gamma$ , we obtain an edge-preserving smoothing.

Open active contours, i.e. active contours with free endpoints, are also considered by [27], where the authors propose a method for detection of open boundaries based on an edge detector which uses the image gradient. Here, we consider approaches based on the Mumford-Shah model. Using convex relaxation approaches, global minimizers of the Mumford-Shah functional are determined in [28]. The method can also handle free endpoints. In [29], the level set method is used for evolving curves with free endpoints. However, two level set functions and artificial regions are needed to describe a curve with free endpoints.

During the evolution of curves, topology changes like splitting or merging can occur, since the number and the topology type of edges and region boundaries is often not known in advance. Using indirect methods like level set and phase field

H. Benninghoff is with the Deutsches Zentrum für Luft- und Raumfahrt (DLR), 82234 Weßling, Germany, email: heike.benninghoff@dlr.de.

H. Garcke is with the Fakultät für Mathematik, Universität Regensburg, 93040 Regensburg, Germany, email: harald.garcke@ur.de.

Manuscript received March 24, 2015; revised month dd, yyyy.

techniques, topology changes are handled automatically. It is often argued that the inability to change the topology of curves is the main disadvantage of parametric methods like the original snake model [1]. In this paper, we extend an efficient method to detect and perform topology changes (presented in [30] and based on the original idea of [31], [32]), such that also topology changes of curves with free endpoints can be handled.

The objective of this article is to solve the Mumford-Shah problem including curves with free endpoints with a parametric approach. The method we propose is based on a parametrization of the evolving curves. We show how a method developed for interface curves [30] can be extended for curves with free endpoints. With the presented concept for image segmentation and restoration, we can easily process images with both open and closed edges. Our method is very efficient from a computational point of view, since the curve evolution problem is a one-dimensional problem and no artificial regions have to be used compared to [29].

## II. IMAGE PROCESSING WITH PARAMETRIC CONTOURS

Let  $u_0 : \Omega \rightarrow \mathbb{R}$  be an image function describing for each point in the image domain  $\Omega \subset \mathbb{R}^2$  the intensity of the image.

The Mumford-Shah method [11] for optimal approximation of images aims at finding a set of curves  $\Gamma = \Gamma_1 \cup \dots \cup \Gamma_{N_C}$  and a piecewise smooth function  $u : \Omega \rightarrow \mathbb{R}$  with possible discontinuities across  $\Gamma$  approximating the original image  $u_0$ . The energy to be minimized is

$$E(u, \Gamma) = \sigma |\Gamma| + \int_{\Omega \setminus \Gamma} \|\nabla u\|^2 dx + \lambda \int_{\Omega} (u_0 - u)^2 dx, \quad (1)$$

where  $\sigma, \lambda > 0$  are weighting parameters and  $|\Gamma|$  denotes the total length of the curves in  $\Gamma$ .

A minimizer of the Mumford-Shah functional provides (i) a restoration of the possible noisy original image by a piecewise smooth approximation  $u$  and (ii) a segmentation of the image given by a union of curves  $\Gamma$  representing the set of edges in the image. The curves belonging to  $\Gamma$  can be sharp edges where the image function rapidly changes, but they can also be so-called weak edges where the image function smoothly changes its value, see [10].

The contours  $\Gamma_i$ ,  $i = 1, \dots, N_C$ , may be closed contours with  $\partial\Gamma = \emptyset$ , or open contours with two endpoints. The endpoints may lie on the image boundary  $\partial\Omega$ , may belong to triple junctions where three curves meet, or may be free endpoints, cf. the conjecture of Mumford and Shah [11]. In the latter case, the endpoint is a point inside the image domain, where no other curve continues. Figure 1 shows an image where an edge occurs which terminates near the image center. The edge can be represented by a curve with one endpoint located at the left image boundary  $\partial\Omega$  and one endpoint being a free endpoint, located close to the image center.

In [30], we proposed a parametric method for image segmentation with piecewise constant image approximations  $u$  and interface curves  $\Gamma_1, \dots, \Gamma_{N_C}$ , each separating two regions. There, we considered a decomposition of the image in  $N_R$  regions  $\Omega_1, \dots, \Omega_{N_R}$  separated by curves  $\Gamma_i$ ,  $i = 1, \dots, N_C$ ,

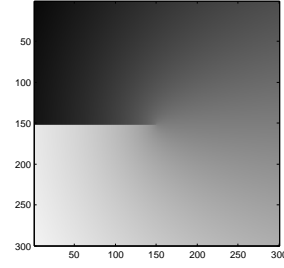


Fig. 1. Image containing an edge with a free endpoint.

and approximations  $u|_{\Omega_k} = c_k \in \mathbb{R}$ . In that case, the functional (1) reduces to

$$E(\Gamma, c_1, \dots, c_{N_R}) = \sigma |\Gamma| + \lambda \sum_{k=1}^{N_R} \int_{\Omega_k} (u_0 - c_k)^2 dx, \quad (2)$$

see [10]. Using methods from the theory of calculus of variations, the following evolution equation can be derived for time-dependent curves  $\Gamma_i(t)$ ,  $t \in [0, T]$ :

$$(V_n)_i = \sigma \kappa_i + F_i, \quad i = 1, \dots, N_C, \quad (3)$$

where  $(V_n)_i$  is the normal velocity of  $\Gamma_i(t)$ ,  $\kappa_i$  is the curvature, and  $F_i$  is given by

$$F_i(\cdot, t) = \lambda [(u_0 - c_{k^+(i)}(t))^2 - (u_0 - c_{k^-(i)}(t))^2], \quad (4)$$

with  $t \in [0, T]$ . The indices  $k^+(i), k^-(i) \in \{1, \dots, N_R\}$  denote the two regions which are separated by  $\Gamma_i(t)$ . The coefficients  $c_k(t)$ ,  $k = 1, \dots, N_R$ , are the mean of  $u_0$  in the region  $\Omega_k(t)$ .

In practice, the segmentation problem can be solved in a two-step approach. For discrete time steps  $t \geq 0$ , the coefficients  $c_k$  are computed using the current set of curves. This is followed by an update of the curves  $\Gamma(t) \rightarrow \Gamma(t + \Delta t)$ , performed by solving the evolution equation (3).

For some images, the piecewise constant approximation is not applicable, see the exemplary image in Figure 1. For such images, the image domain cannot be decomposed in regions separated by interface curves.

Consequently, we modify the two-step approach of [30], such that also non-interface curves with free endpoints can be dealt with. In the first step, we will solve a diffusion equation in the image domain resulting in a piecewise smooth approximation  $u$ . Instead of using the coefficients  $c_k$ , we will consider for  $\vec{p} \in \Gamma_i(t)$  the limit  $u^\pm(\vec{p}) = \lim_{\epsilon \rightarrow 0, \epsilon > 0} u(\vec{p} \pm \epsilon \vec{\nu}_i(\vec{p}))$ , where  $\vec{\nu}_i$  is a normal vector field on  $\Gamma_i(t)$ . Having computed  $u$ , we solve the evolution equation (3) with a modified external term  $F_i$ , using  $u^\pm$  instead of constants  $c_{k^\pm(i)}$ .

Before, presenting further details, we first consider the regularity of a solution of the Mumford-Shah problem at the free endpoint. Fixing the curves  $\Gamma$ , let  $u$  denote the minimizer of the Mumford-Shah energy (1). At free endpoints, problems concerning the regularity of  $u$  occur, cf. [11]. Expressed in polar coordinates  $(r, \phi)$  centered at the free endpoint, the solution  $u$  is of the form

$$u(r, \phi) = c r^{1/2} \sin\left(\frac{1}{2}(\phi - \phi_0)\right) + \hat{v}(r, \phi), \quad (5)$$

where  $\hat{v}$  is a  $C^1$ -function and  $c, \phi_0$  are constants, see [33].

For image segmentation, we later need to solve the problem on a discrete set: Let  $\Omega^h$  be a rectangular grid of nodes covering  $\Omega$  with grid size  $h > 0$ . We replace the second integral on the right hand side of (1) by a sum containing difference quotients of the form

$$\nabla_h^i u(\vec{z}) = \frac{1}{h} (u(\vec{z} + h\vec{e}_i) - u(\vec{z})), \quad \vec{z} \in \Omega^h, \quad (6)$$

where  $\vec{e}_i \in \mathbb{R}^2$  are the standard basis vectors of  $\mathbb{R}^2$ ,  $i = 1, 2$ . For image segmentation applications, we choose the pixel grid, i.e. we use  $h = 1$ . For the approximating sum, we have to exclude terms where the line  $[\vec{z}, \vec{z} + h\vec{e}_i]$  intersects with the curve  $\Gamma$ .

Instead of the original Mumford-Shah functional (1), we thus consider the energy

$$\begin{aligned} E^h(\Gamma, u) = & \sigma|\Gamma| + \sum_{\substack{\vec{z} \in \Omega^h \\ \text{s.t. } \vec{z} + h\vec{e}_2 \in \Omega^h}} (1 - \alpha_x(\vec{z})) (\nabla_h^2 u(\vec{z}))^2 + \\ & + \sum_{\substack{\vec{z} \in \Omega^h \\ \text{s.t. } \vec{z} + h\vec{e}_1 \in \Omega^h}} (1 - \alpha_y(\vec{z})) (\nabla_h^1 u(\vec{z}))^2 + \lambda \int_{\Omega} (u_0 - u)^2 dx, \end{aligned} \quad (7)$$

where  $\alpha_x(\vec{z}), \alpha_y(\vec{z}) \in [0, 1]$  are scalar terms. If  $[\vec{z}, \vec{z} + h\vec{e}_1]$  intersects with  $\Gamma$ ,  $\alpha_y(\vec{z})$  is set to 1.

#### A. Example

We consider one single open curve  $\Gamma$ . Let  $\vec{x} : [0, 1] \rightarrow \mathbb{R}^2$  with  $\vec{x}([0, 1]) = \Gamma$  be a parameterization of the curve. Let  $\vec{x}(0)$  be a free endpoint and let  $\vec{x}(1)$  intersect with the image boundary.

Figure 2 visualizes a possible situation near the free endpoint  $\vec{x}(0)$ . Let  $\vec{z}_{++}, \vec{z}_{+-}, \vec{z}_{--}, \vec{z}_{-+}$  denote the four grid points around  $\vec{x}(0)$  as shown in Figure 2. In this example, the tangential vector of the curve at  $\vec{x}(0)$  is  $\vec{\tau}(0) = \vec{x}_s(0) = \vec{e}_1$ , where  $s$  denotes the arc-length of the curve.

Considering  $\vec{z} = \vec{z}_{+-}$ , the line  $[\vec{z}_{+-}, \vec{z}_{++}]$  and  $\Gamma$  intersect. Thus,  $\alpha_x(\vec{z}_{+-})$  is set to 1. For  $\vec{z} = \vec{z}_{--}$ , we define a factor

$$\alpha_x(\vec{z}_{--}) := \frac{1}{h} ((\vec{z}_{+-})_1 - (\vec{x}(0))_1), \quad (8)$$

where  $(\cdot)_i$  denotes the  $i$ -th component of a vector,  $i = 1, 2$ . The factor  $\alpha_x(\vec{z}_{--})$  describes how far the curve has entered the square given by  $\vec{z}_{++}, \vec{z}_{+-}, \vec{z}_{--}, \vec{z}_{-+}$ .

For  $\vec{z} \in \Omega^h$ ,  $\vec{z} \neq \vec{z}_{--}$ , we set  $\alpha_x(\vec{z}) = 0$  if  $[\vec{z}, \vec{z} + h\vec{e}_2] \cap \Gamma = \emptyset$  and  $\alpha_x(\vec{z}) = 1$  else. The factor  $\alpha_y(\vec{z})$  is defined similarly.

We now want to vary  $\Gamma$  in direction  $-\vec{\tau}(0)$  at  $\vec{x}(0)$ . We consider a second curve  $\Gamma^\epsilon$ ,  $\epsilon > 0$ , with a parameterization  $\vec{x}^\epsilon$ , such that  $\vec{x}^\epsilon(0) = \vec{x}(0) - \epsilon\vec{\tau}(0)$ . We can assume that  $\epsilon$  is small enough, such that  $\vec{x}^\epsilon(0)$  is still inside the square given by  $\vec{z}_{++}, \vec{z}_{+-}, \vec{z}_{--}, \vec{z}_{-+}$ .

The energy difference is

$$\begin{aligned} E^h(\Gamma^\epsilon, u) - E^h(\Gamma, u) = & \sigma\epsilon + (1 - \alpha_x(\vec{z}_{--}) - \epsilon)(\nabla_h^2 u(\vec{z}_{--}))^2 \\ & - (1 - \alpha_x(\vec{z}_{--}))(\nabla_h^2 u(\vec{z}_{--}))^2 \\ = & \sigma\epsilon - \epsilon(\nabla_h^2 u(\vec{z}_{--}))^2. \end{aligned}$$

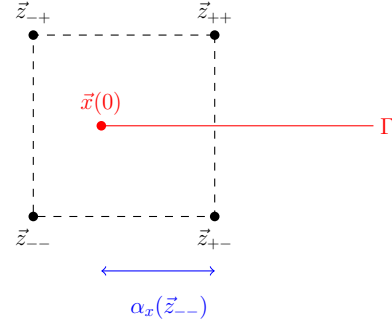


Fig. 2. Illustration of the pixel grid close to the free endpoint.

The energy will decrease if

$$\sigma < (\nabla_h^2 u(\vec{z}_{--}))^2. \quad (9)$$

Thus a motion of a curve in direction  $-\vec{\tau}(0) = -\vec{e}_1$  at the free endpoint  $\vec{x}(0)$  requires that the square of the difference quotient of  $u$  in  $\vec{e}_2$ -direction at  $\vec{z}_{--}$  is sufficient large compared to the weighting parameter  $\sigma$  of the length term in the energy (7).

#### B. General Case

We consider a curve with one or two free endpoints with tangential vector  $\vec{\tau}(\rho)$  at the free endpoint  $\vec{x}(\rho)$ ,  $\rho \in \{0, 1\}$ . We define the factors  $\alpha_x$  and  $\alpha_y$  as follows: Let  $\vec{z}_{++}(\rho), \vec{z}_{+-}(\rho), \vec{z}_{--}(\rho), \vec{z}_{-+}(\rho)$  denote the four grid points around  $\vec{x}(\rho)$ .

If  $\rho = 0$  and  $\vec{\tau}(0) \cdot \vec{e}_1 \geq 0$  or if  $\rho = 1$  and  $\vec{\tau}(1) \cdot \vec{e}_1 < 0$ , we define  $\vec{z}^{p,1} := \vec{z}_{--}(\rho)$  and set

$$\alpha_x(\vec{z}^{p,1}) := 1 - \frac{1}{h} ((\vec{x}(\rho))_1 - (\vec{z}^{p,1})_1).$$

If  $\rho = 0$  and  $\vec{\tau}(0) \cdot \vec{e}_1 < 0$  or if  $\rho = 1$  and  $\vec{\tau}(1) \cdot \vec{e}_1 \geq 0$ , we define  $\vec{z}^{p,1} := \vec{z}_{+-}(\rho)$  and set

$$\alpha_x(\vec{z}^{p,1}) := 1 - \frac{1}{h} ((\vec{z}^{p,1})_1 - (\vec{x}(\rho))_1).$$

If  $\rho = 0$  and  $\vec{\tau}(0) \cdot \vec{e}_2 \geq 0$  or if  $\rho = 1$  and  $\vec{\tau}(1) \cdot \vec{e}_2 < 0$ , we define  $\vec{z}^{p,2} := \vec{z}_{--}(\rho)$  and set

$$\alpha_y(\vec{z}^{p,2}) := 1 - \frac{1}{h} ((\vec{x}(\rho))_2 - (\vec{z}^{p,2})_2).$$

If  $\rho = 0$  and  $\vec{\tau}(0) \cdot \vec{e}_2 < 0$  or if  $\rho = 1$  and  $\vec{\tau}(1) \cdot \vec{e}_2 \geq 0$ , we define  $\vec{z}^{p,2} := \vec{z}_{-+}(\rho)$  and set

$$\alpha_y(\vec{z}^{p,2}) := 1 - \frac{1}{h} ((\vec{z}^{p,2})_2 - (\vec{x}(\rho))_2).$$

Using these definitions, we define the following factors for  $\vec{z} \in \Omega^h$ :

$$\alpha_x(\vec{z}) = \begin{cases} 1, & \text{if } [\vec{z}, \vec{z} + h\vec{e}_2] \cap \Gamma \neq \emptyset, \text{ and } \\ & \vec{z} \neq \vec{z}^{p,1}, \rho \in \{0, 1\}, \\ \alpha_x(\vec{z}^{p,1}), & \text{if } \vec{z} = \vec{z}^{p,1}, \rho \in \{0, 1\}, \\ 0, & \text{else.} \end{cases}$$

and

$$\alpha_y(\vec{z}) = \begin{cases} 1, & \text{if } [\vec{z}, \vec{z} + h\vec{e}_1] \cap \Gamma \neq \emptyset, \text{ and} \\ & \vec{z} \neq \vec{z}^{p,2}, \rho \in \{0, 1\}, \\ \alpha_y(\vec{z}^{p,2}), & \text{if } \vec{z} = \vec{z}^{p,2}, \rho \in \{0, 1\}, \\ 0, & \text{else.} \end{cases}$$

For minimizing (7), we propose the following approach:

Assume the case of one curve  $\Gamma$  with two free endpoints parameterized by  $\vec{x} : [0, 1] \rightarrow \mathbb{R}^2$ . In the first step, we fix  $u$  in (7) and consider for  $\vec{\eta} : [0, 1] \rightarrow \mathbb{R}^2$  a variation of  $\vec{x}$  of the form  $\vec{x} + \epsilon\vec{\eta}$ ,  $\epsilon > 0$ .

Let  $\Gamma^{\epsilon, \eta}$  denote the image of  $\vec{x} + \epsilon\vec{\eta}$ . We use the notation  $E^h(\vec{\eta}) := E^h(\Gamma^{\epsilon, \eta}, u)$  and compute

$$\begin{aligned} \left. \frac{d}{d\epsilon} \right|_{\epsilon=0} E^h(\vec{\eta}) &= \left. \frac{d}{d\epsilon} \right|_{\epsilon=0} E^h(\Gamma^{\epsilon, \eta}, u) \\ &= -\sigma \int_{\Gamma} \vec{x}_{ss} \cdot \vec{\eta} ds - \int_{\Gamma} F \vec{\nu} \cdot \vec{\eta} ds + \\ &\quad + \sigma \vec{x}_s(1) \cdot \vec{\eta}(1) - \sigma \vec{x}_s(0) \cdot \vec{\eta}(0) \\ &\quad - \text{sign}(\vec{\tau}(1) \cdot \vec{e}_1) \vec{\eta}(1) \cdot \vec{e}_1 (\nabla_h^2 u(\vec{z}^{1,1}))^2 \\ &\quad - \text{sign}(\vec{\tau}(1) \cdot \vec{e}_2) \vec{\eta}(1) \cdot \vec{e}_2 (\nabla_h^1 u(\vec{z}^{1,2}))^2 \\ &\quad + \text{sign}(\vec{\tau}(0) \cdot \vec{e}_1) \vec{\eta}(0) \cdot \vec{e}_1 (\nabla_h^2 u(\vec{z}^{0,1}))^2 \\ &\quad + \text{sign}(\vec{\tau}(0) \cdot \vec{e}_2) \vec{\eta}(0) \cdot \vec{e}_2 (\nabla_h^1 u(\vec{z}^{0,2}))^2. \end{aligned}$$

For this computation, integration by parts and a transport theorem are applied. Further,  $\vec{\nu}$  is a normal vector field on  $\Gamma$  such that the pair  $(\vec{x}_s, \vec{\nu})$  is a positive oriented basis of  $\mathbb{R}^2$  and  $F$  is defined as the jump

$$F = \lambda[(u_0 - u^+)^2 - (u_0 - u^-)^2]. \quad (10)$$

We define the following inner product for functions  $\vec{\eta}, \vec{\chi} : [0, 1] \rightarrow \mathbb{R}^2$ :

$$(\vec{\eta}, \vec{\chi})_{2, \Gamma, \partial \Gamma} := \int_{\Gamma} \vec{\eta} \cdot \vec{\chi} ds + \vec{\eta}(1) \cdot \vec{\chi}(1) + \vec{\eta}(0) \cdot \vec{\chi}(0). \quad (11)$$

Now, we consider a family of curves  $\Gamma(t)$ ,  $t \in [0, T]$ . Let  $\vec{x} : [0, 1] \times [0, T] \rightarrow \mathbb{R}^2$  be a mapping such that  $\vec{x}(\cdot, t)$  is a parameterization of  $\Gamma(t)$ ,  $t \in [0, T]$ . We call  $\vec{x}$  a solution of the gradient flow equation, if

$$(\vec{x}_t, \vec{\eta})_{2, \Gamma, \partial \Gamma} = - \left. \frac{d}{d\epsilon} \right|_{\epsilon=0} E^h(\vec{\eta}) \quad (12)$$

holds for all  $\vec{\eta} : [0, 1] \rightarrow \mathbb{R}^2$ .

In the following, we consider particular choices of functions  $\vec{\eta}$  and derive evolution equations for the curve. First, we consider  $\vec{\eta} = \eta_0 \vec{\nu}$  for a scalar function  $\eta_0 : [0, 1] \rightarrow \mathbb{R}$  with  $\eta_0(0) = \eta_0(1) = 0$ . This provides

$$\int_{\Gamma(t)} \vec{x}_t \cdot \vec{\nu} \eta_0 ds = \int_{\Gamma(t)} (\sigma \vec{x}_{ss} \cdot \vec{\nu} + F) \eta_0 ds.$$

Since  $\eta_0$  is arbitrary chosen (with value 0 at the endpoints), we conclude the following equation for the normal velocity of the curve:

$$V_n := \vec{x}_t \cdot \vec{\nu} = \sigma \kappa + F, \quad (13a)$$

using the identity

$$\kappa \vec{\nu} = \vec{x}_{ss}. \quad (13b)$$

Next, we choose  $\vec{\eta} = \eta_0 \vec{\tau}$ , where  $\eta_0 : [0, 1] \rightarrow \mathbb{R}$  is a scalar function with  $\eta_0(0) \neq 0$  and  $\eta_0(1) = 0$ , i.e.  $\vec{\eta}(0) = \vec{\tau}(0) \eta_0(0)$  and  $\vec{\eta}(1) = \vec{0}$ . Inserting  $\vec{\eta}$  in (12), and using (13a), (13b) and  $\vec{\tau} = \vec{x}_s$ , leads to

$$\begin{aligned} \vec{x}_t(0) \cdot \vec{\tau}(0) \eta_0(0) &= \sigma \eta_0(0) \\ &\quad - \text{sign}(\vec{\tau}(0) \cdot \vec{e}_1) \vec{\tau}(0) \eta_0(0) \cdot \vec{e}_1 (\nabla_h^2 u(\vec{z}^{0,1}))^2 + \\ &\quad - \text{sign}(\vec{\tau}(0) \cdot \vec{e}_2) \vec{\tau}(0) \eta_0(0) \cdot \vec{e}_2 (\nabla_h^1 u(\vec{z}^{0,2}))^2. \end{aligned}$$

Since  $\eta_0(0)$  is arbitrary and  $\text{sign}(\vec{\tau}(0) \cdot \vec{e}_i) \vec{\tau}(0) \cdot \vec{e}_i = |\vec{\tau}(0) \cdot \vec{e}_i|$  for  $i = 1, 2$ , we conclude for the tangential velocity

$$\begin{aligned} V_{\tan}(0) &:= \vec{x}_t(0) \cdot \vec{\tau}(0) \\ &= \sigma - |\vec{\tau}(0) \cdot \vec{e}_1| (\nabla_h^2 u(\vec{z}^{0,1}))^2 \\ &\quad - |\vec{\tau}(0) \cdot \vec{e}_2| (\nabla_h^1 u(\vec{z}^{0,2}))^2. \end{aligned} \quad (13c)$$

Choosing  $\eta_0(0) = 0$  and  $\eta_0(1) \neq 0$ , we can derive the following equation for the tangential velocity in  $\vec{x}(1)$ :

$$\begin{aligned} V_{\tan}(1) &:= \vec{x}_t(1) \cdot \vec{\tau}(1) \\ &= -\sigma + |\vec{\tau}(1) \cdot \vec{e}_1| (\nabla_h^2 u(\vec{z}^{1,1}))^2 \\ &\quad + |\vec{\tau}(1) \cdot \vec{e}_2| (\nabla_h^1 u(\vec{z}^{1,2}))^2. \end{aligned} \quad (13d)$$

Similarly, choosing  $\vec{\eta} = \eta_0 \vec{\nu}$ , provides the following equations for the normal velocity at the free endpoints:

$$\begin{aligned} V_n(0) &:= \vec{x}_t(0) \cdot \vec{\nu}(0) \\ &= -\text{sign}(\vec{\tau}(0) \cdot \vec{e}_1) \vec{\nu}(0) \cdot \vec{e}_1 (\nabla_h^2 u(\vec{z}^{0,1}))^2 \\ &\quad - \text{sign}(\vec{\tau}(0) \cdot \vec{e}_2) \vec{\nu}(0) \cdot \vec{e}_2 (\nabla_h^1 u(\vec{z}^{0,2}))^2, \end{aligned} \quad (13e)$$

$$\begin{aligned} V_n(1) &:= \vec{x}_t(1) \cdot \vec{\nu}(1) \\ &= +\text{sign}(\vec{\tau}(1) \cdot \vec{e}_1) \vec{\nu}(1) \cdot \vec{e}_1 (\nabla_h^2 u(\vec{z}^{1,1}))^2 \\ &\quad + \text{sign}(\vec{\tau}(1) \cdot \vec{e}_2) \vec{\nu}(1) \cdot \vec{e}_2 (\nabla_h^1 u(\vec{z}^{1,2}))^2. \end{aligned} \quad (13f)$$

The curve  $\Gamma$  will grow locally at  $\vec{x}(0)$ , if the curve moves in direction  $-\vec{\tau}(0)$ . In this case  $V_{\tan}(0) = \vec{x}_t(0) \cdot \vec{\tau}(0) < 0$ . Therefore,

$$\sigma < |\vec{\tau}(0) \cdot \vec{e}_1| (\nabla_h^2 u(\vec{z}^{0,1}))^2 + |\vec{\tau}(0) \cdot \vec{e}_2| (\nabla_h^1 u(\vec{z}^{0,2}))^2 \quad (14)$$

has to be satisfied such that the curve length increases. For the exemplary case  $\vec{\tau}(0) = \vec{e}_1$ , the condition reduces to (9), i.e. to the condition from the introductory example.

The curve  $\Gamma(t)$  will grow at  $\vec{x}(1)$ , if the curve moves in direction  $\vec{\tau}(1)$  leading to  $V_{\tan}(1) > 0$ . Therefore, the inequality

$$\sigma < |\vec{\tau}(1) \cdot \vec{e}_1| (\nabla_h^2 u(\vec{z}^{1,1}))^2 + |\vec{\tau}(1) \cdot \vec{e}_2| (\nabla_h^1 u(\vec{z}^{1,2}))^2 \quad (15)$$

has to be satisfied.

Since the term  $\sigma|\Gamma|$  in the energy (7) penalizes the length of the curve, a curve can only grow in direction  $-\vec{\tau}(0)$  or  $\vec{\tau}(1)$ , if the derivative terms  $(\nabla_h^i u)^2$ ,  $i = 1, 2$ , are large compared to  $\sigma$ .

The scheme (13) describes the motion of the curve. For  $N_C$  curves  $\Gamma_1, \dots, \Gamma_{N_C}$ , we can solve (13) for each curve. For a closed curve  $\Gamma_i$ , only the normal velocity (13a) with the relation (13b) needs to be considered, on noting that  $\vec{x}_i(0) = \vec{x}_i(1)$ . In the case of triple junctions and intersections with the image boundary, additional conditions for the involved

endpoints have to be considered. If triple junctions occur, the evolution equations for the corresponding three curves which meet at the junction are coupled. The cases with triple junctions and boundary intersection points are described in [30] in detail.

We alternately solve the scheme of evolution equations (13) and recompute the approximating function  $u$  using the updated curve set. The function  $u$  is obtained by solving a diffusion equation on  $\Omega^h$ . We note that (7) is formulated for a discrete set  $\Omega^h$ . We will describe in the next section, how  $u$  is computed numerically.

### III. NUMERICAL APPROXIMATION

#### A. Numerical Solution of the Evolution Equations

For computing the position of the evolving curves  $\Gamma$  numerically, we consider a decomposition of the interval  $[0, 1]$  of the form  $0 = q_0^i < q_1^i < \dots < q_{N_i}^i = 1$ , for  $i = 1, \dots, N_C$ . If  $\Gamma_i$  is a closed curve, we make use of the periodicity  $N_i = 0$ ,  $N_i + 1 = 1$ ,  $-1 = N_i - 1$ , etc.

Further, let  $0 = t_0 < t_1 < \dots < t_M = T$  be a partitioning of the time interval  $[0, T]$  with time steps  $\Delta t_m := t_{m+1} - t_m$ ,  $m = 0, \dots, M - 1$ . Smooth curves  $\Gamma_i(t_m)$ ,  $i = 1, \dots, N_C$ ,  $m = 0, \dots, M$  are replaced by polygonal curves  $\Gamma_i^m$  given by nodes  $\vec{X}_{i,j}^m$  which are approximations of  $\vec{x}_i(q_j^i, t_m)$ . Further, let  $\kappa_{i,j}^m$  be an approximation of  $\kappa_i(q_j^i, t_m)$ . The derivative terms with respect to time are replaced by difference quotients of the form

$$(\vec{x}_i)_t(q_j^i, t_m) \approx \frac{1}{\Delta t_m} (\vec{X}_{i,j}^{m+1} - \vec{X}_{i,j}^m). \quad (16)$$

Let  $h_{i,j-\frac{1}{2}}^m = \vec{X}_{i,j}^m - \vec{X}_{i,j-1}^m$ ,  $i = 1, \dots, N_C$ ,  $j = 1, \dots, N_i$ , be the distance between two neighboring nodes. For each curve, we define a discrete normal vector field  $\vec{\nu}_i^m$  by

$$\vec{\nu}_i^m|_{[q_{j-1}^i, q_j^i]} := \vec{\nu}_{i,j-\frac{1}{2}}^m := \frac{(\vec{X}_{i,j}^m - \vec{X}_{i,j-1}^m)^\perp}{h_{i,j-\frac{1}{2}}^m},$$

see also [30], [34], [35]. Here,  $\perp$  denotes the anti-clockwise rotation of a vector by  $\pi/2$ . Further, we define the following weighted approximating normal vector at  $\vec{X}_{i,j}^m$  by

$$\vec{\omega}_{i,j}^m := \frac{h_{i,j-\frac{1}{2}}^m \vec{\nu}_{i,j-\frac{1}{2}}^m + h_{i,j+\frac{1}{2}}^m \vec{\nu}_{i,j+\frac{1}{2}}^m}{h_{i,j-\frac{1}{2}}^m + h_{i,j+\frac{1}{2}}^m} = \frac{(\vec{X}_{i,j+1}^m - \vec{X}_{i,j-1}^m)^\perp}{h_{i,j-\frac{1}{2}}^m + h_{i,j+\frac{1}{2}}^m},$$

for  $j = 1, \dots, N_i$  if  $\partial\Gamma_i^m = \emptyset$  and for  $j = 1, \dots, N_i - 1$  if  $\partial\Gamma_i^m \neq \emptyset$ . In the latter case, we set

$$\begin{aligned} \vec{\omega}_{i,0}^m &:= \vec{\nu}_{i,\frac{1}{2}}^m = \frac{(\vec{X}_{i,1}^m - \vec{X}_{i,0}^m)^\perp}{h_{i,\frac{1}{2}}^m}, \\ \vec{\omega}_{i,N_i}^m &:= \vec{\nu}_{i,N_i-\frac{1}{2}}^m = \frac{(\vec{X}_{i,N_i}^m - \vec{X}_{i,N_i-1}^m)^\perp}{h_{i,N_i-\frac{1}{2}}^m}. \end{aligned}$$

The external term  $F$  is approximated by

$$\begin{aligned} F_{i,j}^m &:= \lambda \left[ (u_0(\vec{X}_{i,j}^m) - u(\vec{X}_{i,j}^m + a\vec{\omega}_{i,j}^m))^2 \right. \\ &\quad \left. - (u_0(\vec{X}_{i,j}^m) - u(\vec{X}_{i,j}^m - a\vec{\omega}_{i,j}^m))^2 \right], \end{aligned}$$

with a small real number  $a > 0$  if  $j$  is not the index of a free endpoint, and we set  $F_{i,j}^m = 0$  else.

The equation for the normal velocity (13a) is approximated by

$$\frac{1}{\Delta t_m} (\vec{X}_{i,j}^{m+1} - \vec{X}_{i,j}^m) \cdot \vec{\omega}_{i,j}^m = \sigma \kappa_{i,j}^{m+1} + F_{i,j}^m. \quad (17a)$$

Thus, for computing  $\Gamma_i^{m+1}$ , we use the previous curve  $\Gamma_i^m$  for the external term  $F_{i,j}^m$  and for the weighted normal  $\vec{\omega}_{i,j}^m$ .

For an approximation of (13b), we need to define an approximation of  $(\vec{x}_i)_{ss}(q_j^i, t_{m+1})$ . For that, we make use of difference quotients of the form

$$\begin{aligned} \Delta_2^{h,m} \vec{X}_{i,j}^{m+1} &:= \frac{2}{h_{i,j-\frac{1}{2}}^m + h_{i,j+\frac{1}{2}}^m} \left( (\vec{X}_{i,j+1}^{m+1} - \vec{X}_{i,j}^{m+1})/h_{i,j+\frac{1}{2}}^m \right. \\ &\quad \left. - (\vec{X}_{i,j}^{m+1} - \vec{X}_{i,j-1}^{m+1})/h_{i,j-\frac{1}{2}}^m \right), \end{aligned}$$

for  $i = 1, \dots, N_C$  and  $j = 1, \dots, N_i$ , if  $\partial\Gamma_i^m = \emptyset$ , and  $j = 1, \dots, N_i - 1$ , else. In case of equal spatial step sizes  $h_{i,j-\frac{1}{2}}^m = h_{i,j+\frac{1}{2}}^m =: h_i^m$ , the term reduces to  $(\vec{X}_{i,j-1}^m - 2\vec{X}_{i,j}^m + \vec{X}_{i,j+1}^m)/(h_i^m)^2$ , see also [30], where we also defined and used these difference quotients.

The equation (13b) is now approximated by

$$\kappa_{i,j}^{m+1} \vec{\omega}_{i,j}^m = \Delta_2^{h,m} \vec{X}_{i,j}^{m+1}, \quad (17b)$$

for  $i = 1, \dots, N_C$ ,  $j = 1, \dots, N_i$  in case of closed curves and  $j = 1, \dots, N_i - 1$  in case of open curves.

In case of open curves, additional equations for the endpoints are needed. The case of triple junctions and boundary intersection points is described in [30]. For curves with free endpoints we introduce the tangential vectors  $\vec{\tau}_{i,0}^m = (\vec{X}_{i,1}^m - \vec{X}_{i,0}^m)/h_{i,\frac{1}{2}}^m$ , and  $\vec{\tau}_{i,N_i}^m = (\vec{X}_{i,N_i}^m - \vec{X}_{i,N_i-1}^m)/h_{i,N_i-\frac{1}{2}}^m$ . The equations (13c) and (13d) are approximated by

$$\begin{aligned} \frac{1}{\Delta t_m} (\vec{X}_{i,0}^{m+1} - \vec{X}_{i,0}^m) \cdot \vec{\tau}_{i,0}^m &= \\ = \sigma - |\vec{\tau}_{i,0}^m \cdot \vec{e}_1| (\nabla_h^2 u(\vec{z}^{0,1}))^2 - |\vec{\tau}_{i,0}^m \cdot \vec{e}_2| (\nabla_h^1 u(\vec{z}^{0,2}))^2, \end{aligned} \quad (17c)$$

and

$$\begin{aligned} \frac{1}{\Delta t_m} (\vec{X}_{i,N_i}^{m+1} - \vec{X}_{i,N_i}^m) \cdot \vec{\tau}_{i,N_i}^m &= \\ = -\sigma + |\vec{\tau}_{i,N_i}^m \cdot \vec{e}_1| (\nabla_h^2 u(\vec{z}^{1,1}))^2 + |\vec{\tau}_{i,N_i}^m \cdot \vec{e}_2| (\nabla_h^1 u(\vec{z}^{1,2}))^2. \end{aligned} \quad (17d)$$

Similarly, discrete versions of (13e) and (13f) can be stated using  $\vec{\omega}_{i,0}^m$  and  $\vec{\omega}_{i,N_i}^m$  as discrete normal vectors.

The scheme (17) is a numerical approximation of the scheme (13), where the parametric curves are replaced by polygonal curves, and the smooth functions  $\vec{x}_i$  and  $\kappa_i$  are replaced by continuous functions uniquely given by their values at the nodes  $q_j^i$ ,  $i = 1, \dots, N_C$ ,  $j = 0, \dots, N_i$ .

The discrete scheme can be rewritten to a linear system with a sparse system matrix, similar as presented in [30], and can be solved with a fast direct solver like for example the UMFPACK algorithm [36].

### B. Numerical Solution of the Denoising Problem

For computing a numerical solution  $u^h$  for the piecewise smooth, denoised version  $u$  of  $u_0$ , we consider for  $N_x, N_y \in \mathbb{N}$  the discrete set

$$\Omega^h := \{(ih, jh) : i = 0, \dots, N_x, j = 0, \dots, N_y\},$$

where  $N_x$  and  $N_y$  are the number of pixels in  $x$ - and  $y$ -direction. We define for  $i = 1, \dots, N_x, j = 1, \dots, N_y$

$$A_x(i, j) = \begin{cases} h^2, & \text{if } [(i-1)h, ih] \times \{j\} \cap \Gamma_{i_0}^m = \emptyset, \\ & \forall i_0 \in \{1, \dots, N_C\}, \\ 0, & \text{else,} \end{cases}$$

$$A_y(i, j) = \begin{cases} h^2, & \text{if } \{i\} \times [(j-1)h, jh] \cap \Gamma_{i_0}^m = \emptyset, \\ & \forall i_0 \in \{1, \dots, N_C\}, \\ 0, & \text{else.} \end{cases}$$

Fixing the set of curves  $\Gamma$ , we consider the following discrete energy:

$$\begin{aligned} E_{\text{discr}}(u^h) = & \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left( A_x(i, j) \left( \frac{u_{i,j}^h - u_{i-1,j}^h}{h} \right)^2 \right. \\ & + A_y(i, j) \left( \frac{u_{i,j}^h - u_{i,j-1}^h}{h} \right)^2 \Big) \\ & + \lambda \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} h^2 (u_0(ih, jh) - u_{i,j}^h)^2, \quad (18) \end{aligned}$$

which is a discrete analogue of  $\int_{\Omega \setminus \Gamma} (\|\nabla u\|^2 dx + \int_{\Omega} \lambda(u_0 - u)^2 dx$ . Here  $u_{i,j}^h$  approximates  $u$  at the node  $(ih, jh)$ . The piecewise continuous function  $u^h$  is uniquely given by its value at the points in  $\Omega^h$ .

By setting the terms  $A_x(i, j)$  or  $A_y(i, j)$  to zero at points where the line  $[(i-1)h, ih] \times \{j\}$  or  $\{i\} \times [(j-1)h, jh]$  intersects with one of the curves, we approximate the integral over the set  $\Omega \setminus \Gamma$ .

Taking the derivative of the right hand side of (18) with respect to  $u_{i,j}^h$  and setting the resulting term to zero, leads to a linear system. The corresponding system matrix is sparse since each node  $(ih, jh) \in \Omega^h$  is only coupled to a few neighboring nodes. The resulting linear system can be solved with a fast direct or iterative solver by employing the sparse matrix structure.

Considering  $h \rightarrow 0$ , we obtain in the limit  $\nabla u \cdot \vec{\nu} = 0$  at the curves belonging to  $\Gamma$ , and  $\nabla u \cdot \vec{n}_{\partial\Omega} = 0$  at the image boundary  $\partial\Omega$ , where  $\vec{n}_{\partial\Omega}$  is a normal vector field at  $\partial\Omega$ . For details, we refer to [30]. Consequently, we obtain an edge preserving image smoothing if  $\Gamma$  matches with the edges in the given image.

### C. Topology Changes

During the evolution of curves, topology changes can occur, since the edge set in the image and the boundaries of objects are not known in advance. Therefore, curves can split into

two or more subcurves, curves can merge to one single curve, triple junctions and new curves may occur and curves can intersect with the image boundary such that new boundary nodes emerge. Further, a curve needs to be deleted if its length becomes too small. In [30], we extended the idea of [31], [32], and described a method to detect topology changes of curves efficiently. The main idea is the use of an artificial background grid which covers the entire image domain  $\Omega$ . We consider successively all nodes  $\vec{X}_{i,j}^m$  and mark a grid element with  $(i, j)$  if  $\vec{X}_{i,j}^m$  is the first node located in this array. If a grid element is already marked with  $(i_1, j_1)$  and the nodes  $\vec{X}_{i,j}^m$  and  $\vec{X}_{i_1,j_1}^m$  are not neighbor nodes, a topology change likely occurs close to this pair. Details on this method for curves without free endpoints are given in [30].

In principle, topology changes involving curves with free endpoints can be detected similarly by using such a background grid. In addition to the topology changes listed above (splitting, merging, emergence of triple junctions and boundary intersection points), topology changes involving the free endpoints can occur: If two free endpoints of one curve are located in one square of the background grid, an open contour becomes a closed contour. If two free endpoints of two different curves meet, the two curves merge to one single curve, and the former free endpoints become inner nodes of the new curve. If a free endpoint and an inner point of a curve meet, a triple junction is created.

### D. Summary of the Algorithm

We propose the following algorithm for image segmentation and image restoration with parametric contours with possible free endpoints:

Given a set of polygonal curves  $\Gamma^0 = (\Gamma_1^0, \dots, \Gamma_{N_C}^0)$  and  $\vec{X}^0 = (\vec{X}_1^0, \dots, \vec{X}_{N_C}^0)$  with  $\vec{X}_i^0([0, 1]) = \Gamma_i^0$ , perform the following steps for  $m = 0, 1, \dots, M-1$ :

- 1) Compute a denoised image approximation  $u^h$  by minimizing (18) (solve the corresponding sparse linear system).
- 2) Compute the external terms  $F_{i,j}^m$  by using the solution  $u^h$  of step 1. Compute  $\vec{X}^{m+1}$  by solving the linear equation derived from the scheme (17).
- 3) Check whether topology changes occur. If so, execute the topology change.

A segmentation of the image is given by the final set of curves  $\Gamma^M$ . An image restoration is given by the image approximation  $u^h$  from the time step  $t_M$ .

### E. Modifications

Step 2 of the algorithm above can additionally be split in two sub-steps: First, we fix the free endpoints and we let the inner nodes of the curve evolve. Then, we let the endpoints evolve according to the above presented discrete scheme.

The main effort of this method compared to the Chan-Vese method for interface curves is that we have to solve a two-dimensional diffusion equation (bulk equation) several times during the segmentation. In the experiments described in the next section, we perform 10 steps of curve evolution followed



by a solution of the bulk equation. Having computed  $u^h$ , we use it for the next 10 curve evolution steps.

As an alternative, we can start the segmentation using interface curves and the image segmentation method described in [30] (based on the Chan-Vese method [10]) with piecewise constant approximations. As a postprocessing step, we can consider the derivatives of the image function in normal direction at the final curves (or the jump of the image function across the curves). We replace interface curves by curves with free endpoints if the derivatives in normal direction are locally very small. For that, we delete those parts of a curve where the derivative is small which results in curves with free endpoints. Next, we compute some steps of the segmentation method with free endpoints to obtain the final contours.

Topology changes occur only in rare cases when using a postprocessing evolution of curves with free endpoints. In most situations, topology changes are already detected in the previous evolution.

#### IV. RESULTS AND DISCUSSION

The method for image segmentation and restoration presented in sections II and III are applied on some exemplary test images. For all experiments presented in this section, we use constant time steps sizes  $\Delta t_m = \Delta t$ ,  $m = 0, \dots, M-1$ .

In the first experiment, we consider an example where a contour with two free endpoints evolves in the image domain and detects an edge. Figure 3 presents the results of image segmentation and denoising. It can be observed that the image is not smoothed out across the curve  $\Gamma$ . Further a growth of the curve in tangential direction can be observed. The growth stops when the inequalities (14) and (15) become equalities. This depends on the absolute values of the difference quotients  $|\nabla_h^i u|$ ,  $i = 1, 2$ , and the weighting parameter  $\sigma$ . The image approximation  $u$  attains values in  $[0, 1]$ . In this image, differences of the form  $u(\vec{x} + h\vec{e}_i) - u(\vec{x})$  are typically of magnitude  $10^{-2}$ . Since  $\Omega = [1, 300] \times [1, 300]$  and  $h = 1$ ,  $|\nabla_h^i u|^2$  is of magnitude  $10^{-4}$ . Therefore, we have to choose a small value for the weighting parameter  $\sigma$ , here, we choose  $\sigma = 2e - 5$ . If we used a normalized image domain  $\Omega = [0, 1] \times [0, 1]$ , the pixel grid would have a grid size of  $h = 1/300$  and  $h^2 = 1/90000$ . In this case, we would choose a weight  $\sigma$  of magnitude 1.

In a second experiment, we study a crack tip problem which has also been considered in [28]. The image function is given by

$$u_0(\vec{x}) = a\sqrt{r(\vec{x})}\sin(\theta(\vec{x})/2) + b, \quad (19)$$

where  $r(\vec{x}) \geq 0$ ,  $\theta(\vec{x}) \in (-\pi, \pi]$  are polar coordinates with  $r = 0$  corresponding to the image center, and  $a, b \in \mathbb{R}$  are constants such that  $u_0$  attains values in  $[0, 1]$ .

Figure 4 shows the evolution of a contour with one free endpoint. The second endpoint belongs to the image boundary. At time step  $m = 3000$ , the free endpoint is located at the image center and the curve matches with the edge in the image. As discussed above (see also Equations (14) and (15)), the parameter  $\sigma$ , which weights the length term, needs to be chosen small enough such that the curve can extend. If  $\sigma$  is fixed, the absolute value of the difference quotients must be

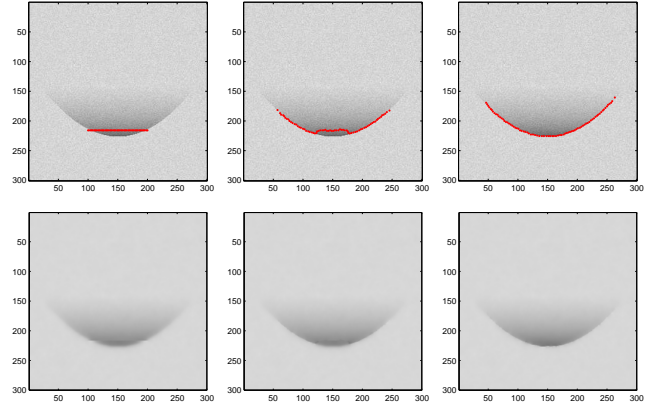


Fig. 3. Example image showing a contour with two free endpoints. Original image and evolving contours (1st row) and denoised image (2nd row) for  $m = 1, 1000, 6000$  using  $\Delta t = 0.032$ ,  $\sigma = 2e - 5$  and  $\lambda = 0.002$ .

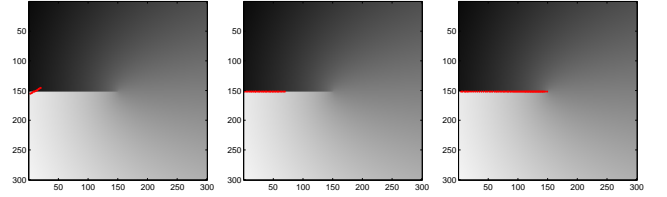


Fig. 4. Example image showing a contour with one free endpoint and one boundary intersection point, see also [28]. Evolving contours for  $m = 1, 500, 3000$  using  $\Delta t = 0.001$ ,  $\sigma = 2e - 5$  and  $\lambda = 0.002$ .

large enough such that the length of the curve increases. In this example, the edge is a horizontal line and the position where the curve stops depends on the value of  $\nabla_h^2 u$ , i.e. on the difference quotient in  $y$ -direction.

We rerun the example using  $\sigma = 0.002$  and  $\sigma = 0.01$  instead of  $\sigma = 2e - 5$ . Figure 5 shows the results at time step  $m = 3000$ . In both cases, the free endpoint does not reach the center of the image since the value of  $\sigma$  has been set larger. The growth of the curve already stops at larger values of  $\nabla_h^2 u$ , recall conditions (14) and (15)). We even let the curve evolve until time step  $m = 5000$ , but we observed no significant motion between  $m = 3000$  and  $m = 5000$ .

In another experiment, which demonstrates the evolution of curves with free endpoints, we first apply the parametric method of [30] to the Chan-Vese problem [10] using interface-curves. This means, that we first segment a given image in regions separated by interface curves, see Figure 6. We start with one large initial curve which splits up in two sub-curves. This example thus also demonstrates the handling of a topology change.

In a postprocessing step, we delete those nodes where the jump of  $u_0$  across the curve is smaller than a given tolerance of  $tol = 0.1$ . This results in one closed curve, where no points are deleted (blue curve in Figure 7), and in one curve with two free endpoints (red curve). Figure 7 shows the results of a postprocessing evolution of the curve. This example shows that our methods for image segmentation and denoising can be applied also on images with both open and closed edges.

Table I shows the values of the discrete Mumford-Shah

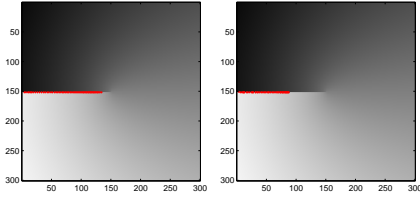


Fig. 5. Dependency on the weighting parameter using  $\sigma = 0.002$  (left) and  $\sigma = 0.01$  (right),  $\Delta t = 0.001$  and  $\lambda = 0.002$  for  $m = 3000$ . If a too large weight is chosen for the length term in (7), the curve does not reach the image center.

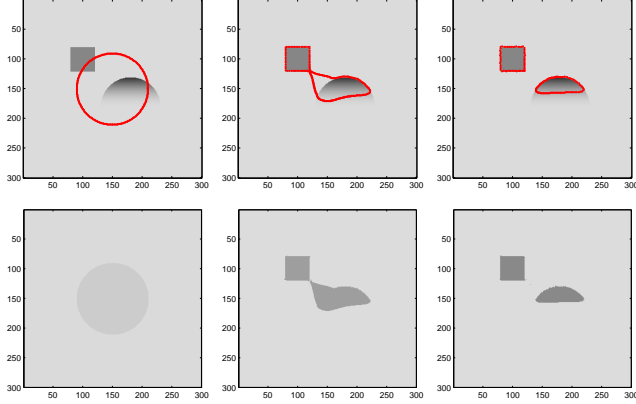


Fig. 6. Image segmentation result using Chan-Vese, interface curves and a piecewise constant image approximation. Original image and contours (first row) and piecewise constant approximation (second row) for  $m = 1, 1050, 1200$ .

TABLE I  
COMPARISON OF DISCRETE MUMFORD-SHAH ENERGY

Processing Method	Step Nr	Discrete Mumford-Shah Energy
Chan-Vese, piecewise constant	1200 (final)	22364.94
Postprocessing, free endpoints	1 (start)	23162.04
Postprocessing, free endpoints	400 (final)	18284.36

energy (7) for the last step of the Chan-Vese piecewise constant segmentation with closed region boundaries (cf. Figure 6,  $m = 1200$ ) and for the initial and final step of the postprocessing with one open boundary (cp. Figure 7,  $m = 1$  and  $m = 400$ ). Note, that the absolute values are large, since the image consists of 90000 pixels and the size of each pixel is  $1 \times 1$ . The average contribution of each pixel to the energy is  $< 1$ ; however, it sums up to a value of magnitude  $2 \cdot 10^4$ .

From Table I, we can observe that the energy even slightly increases from the last step of the Chan-Vese piecewise constant method to the first step of the postprocessing evolution. Deleting part of the curve does *not* decrease the energy in this example. However, at the end of the postprocessing evolution with one open contour, we obtain a decrease of the energy. The energy at  $m = 400$  of the postprocessing is 81.75% of the

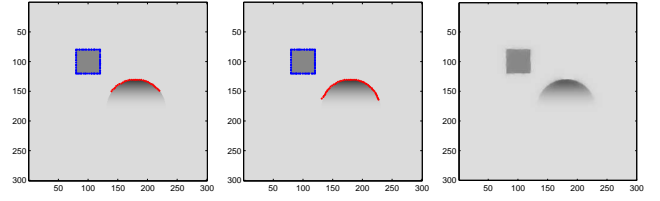


Fig. 7. Postprocessing evolution with a contour with two free endpoints using  $tol = 0.1$  to obtain the initial contour. Original image and contours (left, center) for  $m = 1, 400$  using  $\Delta t = 0.05$ ,  $\sigma = 2e-5$ ,  $\lambda = 2e-4$  and denoised image for  $m = 400$  (right).

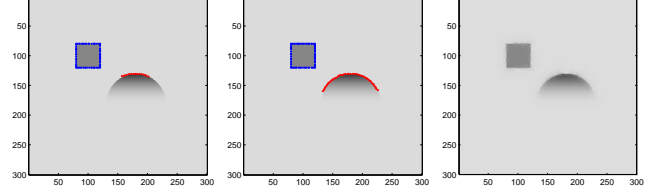


Fig. 8. Postprocessing evolution with a contour with two free endpoints using  $tol = 0.5$  to obtain the initial contour. Original image and contours (left, center) for  $m = 1, 600$  using  $\Delta t = 0.05$ ,  $\sigma = 2e-5$ ,  $\lambda = 2e-4$  and denoised image for  $m = 600$  (right).

energy of the last step of the piecewise constant segmentation with closed boundaries. Therefore, if we delete part of the curve and if we let the curve with free endpoints evolve again, we will obtain a final curve such that the corresponding discrete Mumford-Shah energy (7) is reduced compared to the Chan-Vese piecewise constant result.

Next, we investigate the influence of the tolerance value  $tol$ , which is used for the deletion of some nodes of the curve. Note, that the image function attends values in  $[0, 1]$  where 0 corresponds to black and 1 corresponds to white color. The exact value of the tolerance  $tol$  influences only the start curve of the second curve evolution. We repeat the postprocessing evolution and use  $tol = 0.5$  as tolerance resulting in a different initial curve.

Figure 8 shows the postprocessing evolution of the curve. Of course, since the initial contour in Figure 8 (left) is smaller compared to the initial contour in Figure 7, more iteration steps are needed to obtain the final contour.

The final result is independent on the exact initial curve as long as it is of the same type, i.e. open with free endpoints; not closed or not fully deleted. In our example the largest jump of  $u_0$  across the final red curve of the first evolution (cf. Figure 6, right sub-figures) is 0.61, the smallest jump is 0.05. The large difference between the largest and smallest jump can be used as an indicator to replace the interface-curve by a curve with two free endpoints. (On the contrary, the jump across the blue curve is constant in this example.) As tolerance value  $tol$  any value larger than 0.05 and smaller than 0.61 could be chosen. For  $tol \leq 0.05$  no node point would be deleted resulting in an unchanged curve. For  $tol \geq 0.61$  all nodes and therefore the entire curve would be deleted. All values between the two thresholds can be theoretically used. Therefore, in this example, the final result is independent on the exact value of  $tol$  as long it is in  $(0.05, 0.61)$ .

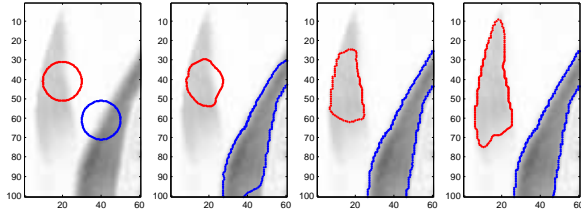


Fig. 9. Medical image segmentation using Chan-Vese, interface curves and a piecewise constant image approximation. Original image and contours for  $m = 1, 400, 1000, 2500$ . Image courtesy: Dr. Declan O'Regan and the Robert Steiner MR Unit, MRC Clinical Sciences Centre, Imperial College London.

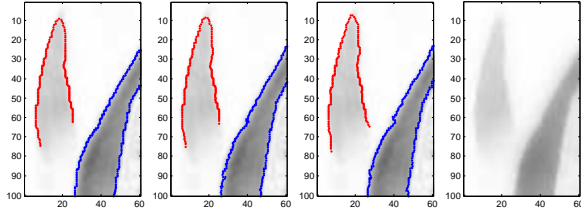


Fig. 10. Postprocessing segmentation with free endpoints. Original image and contours (far left, left, right) for  $m = 1, 100, 500$  using  $\Delta t = 0.008$ ,  $\sigma = 3.3e - 5$ ,  $\lambda = 0.0167$  and denoised image (far right) for  $m = 500$ . Image courtesy: Dr. Declan O'Regan and the Robert Steiner MR Unit, MRC Clinical Sciences Centre, Imperial College London.

Next, we demonstrate an example where a real, medical image is processed. Figure 9 and Figure 10 show an excerpt of a medical image and the result of an edge detection. We first use the Chan-Vese algorithm with piecewise constant image approximation for segmenting the image, see Figure 9. In this first segmentation step, also topology changes occur. The initial closed curve touches twice the image boundary and splits up in two open curves each with two boundary intersection points. After the preceding segmentation, a part of the red curve is deleted (using a tolerance of 0.1 for the jump across the curve) resulting in a curve with free endpoints. Figure 10 shows the result of the postprocessing evolution. Small tangential motions of the free endpoints can be observed.

Finally, we study an example where several topology changes occur. Figure 11 shows an example where we start with many small initial curves. A similar image is also considered in [28]. Many of the small initial lines shrink and are deleted when their curve length becomes too small. Additional topology changes occur: Near the upper left corner of the image, two curves merge at their free endpoints to one curve. Further, three free endpoints become boundary intersection points, and a triple junction emerges when a free endpoint meets another curve at an inner node. The topology changes are detected as described in Section III-C and [30].

An advantage of the parametric method is that we can easily handle non-interface curves and complex curve networks including triple junctions. The curve evolution scheme is very similar to the scheme presented in [30] for interface-curves. Instead of computing the mean value of the image function in regions, we have to solve a diffusion bulk equation. Additional to the motion of the curve in normal direction, free endpoints

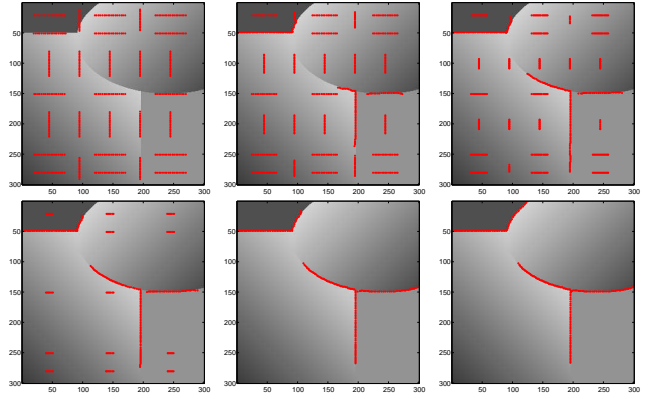


Fig. 11. Image segmentation and contour detection with topology changes. The final segmentation contains two free endpoints, one triple junctions and three boundary intersection points. Original image and contours for  $m = 1, 100, 250, 400, 750, 1200$  using  $\Delta t = 1$ ,  $\sigma = 1e - 4$ ,  $\lambda = 1e - 3$ .

can move in tangential direction.

There are alternatives to parametric methods to describe an evolving curve. The level set method [37] is very popular for image processing applications and in particular for active contours methods, see e.g. [5], [4], [6], [10], [15], [38] to mention a few. In level set methods, a hypersurface is embedded as the zero level set of a function defined on the image domain  $\Omega$ . With level set techniques, free endpoints however cannot be handled with one single level set function: Since level set methods embed a curve as zero level set of a function  $\Phi : \Omega \rightarrow \mathbb{R}$ , the curve is an interface between two regions  $\{\Phi > 0\}$  and  $\{\Phi < 0\}$ . Therefore level sets are always closed or meet the image boundary at their endpoints. Non-interface curves can be handled by using two level set functions  $\Phi$  and  $\Psi$  and by using artificial regions, see [29]. A curve with free endpoints can then be represented by the interface between the artificial regions  $\{\Phi > 0\} \cap \{\Psi > 0\}$  and  $\{\Phi > 0\} \cap \{\Psi < 0\}$ , for example.

Using our direct, parametric approach it is not necessary to introduce artificial regions. Further our method is very efficient, since the curve evolution is only a one-dimensional problem.

## V. CONCLUSION

We proposed a new parametric approach for active contours with free endpoints. The image segmentation and denoising method presented in this article is based on a discrete version of the Mumford and Shah functional. For curves with free endpoints a flow in normal direction and a flow of the endpoints in tangential direction attracts the curves to the edges in the image. With the presented approach, we can handle both open and closed curves. The method is also suitable to be employed as postprocessing step to improve the result of a previous Chan-Vese like segmentation with interface curves and piecewise constant image approximations.

## REFERENCES

- [1] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *Int. J. Comput. Vision*, vol. 1, no. 4, pp. 321–331, 1988.

- [2] L. D. Cohen, "On Active Contour Models and Balloons," *Comput. Vision, Graph. Image Process.: Image Understanding*, vol. 53, no. 2, pp. 211–218, 1991.
- [3] R. Ronfard, "Region-Based Strategies for Active Contour Models," *Int. J. Comput. Vision*, vol. 13, no. 2, pp. 229–251, 1994.
- [4] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic Active Contours," *Int. J. Comput. Vision*, vol. 22, no. 1, pp. 61–79, 1997.
- [5] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape Modeling with Front Propagation: A Level Set Approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 2, pp. 158–175, 1995.
- [6] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi Jr., "Conformal Curvature Flows: From Phase Transitions to Active Vision," *Arch. Ration. Mech. An.*, vol. 134, no. 3, pp. 275–301, 1996.
- [7] S. Araki, N. Yokoya, H. Iwasa, and H. Takemura, "Splitting of Active Contour Models Based on Crossing Detection for Extraction of Multiple Objects," *Syst. Comput. Jpn.*, vol. 28, no. 11, pp. 34–42, 1997.
- [8] N. Paragios and R. Deriche, "Geodesic Active Contours and Level Sets for Detection and Tracking of Moving Objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 3, pp. 266–280, 2000.
- [9] T. F. Chan, B. Y. Sandberg, and L. A. Vese, "Active Contours Without Edges for Vector-Valued Images," *J. Vis. Commun. Image R.*, vol. 11, no. 2, pp. 130–141, 2000.
- [10] T. F. Chan and L. A. Vese, "Active Contours Without Edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, 2001.
- [11] D. Mumford and J. Shah, "Optimal Approximation by Piecewise Smooth Functions and Associated Variational Problems," *Commun. Pur. Appl. Math.*, vol. 42, pp. 577–685, 1989.
- [12] L. Ambrosio and V. M. Tortorelli, "Approximation of Functional Depending on Jumps by Elliptic Functionals via  $\Gamma$ -Convergence," *Commun. Pur. Appl. Math.*, vol. 43, no. 8, pp. 999–1036, 1990.
- [13] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear Total Variation Based Noise Removal Algorithms," *Physica D*, vol. 60, no. 1–4, pp. 259–268, 1992.
- [14] T. F. Chan, S. H. Kang, and J. Shen, "Total Variation Denoising and Enhancement of Color Images Based on the CB and HSV Color Models," *J. Vis. Commun. Image R.*, vol. 12, no. 4, pp. 422–435, 2001.
- [15] A. Tsai, A. Yezzi, and A. S. Willsky, "Curve Evolution Implementation of the Mumford-Shah Functional for Image Segmentation, Denoising, Interpolation and Magnification," *IEEE Trans. Image Process.*, vol. 10, no. 8, pp. 1169–1186, 2001.
- [16] Z. Yu and C. Bajaj, "Anisotropic Vector Diffusion in Image Smoothing," in *Proceedings of International Conference on Image Processing*, 2002, pp. 828–831.
- [17] M. Beneš, V. Chalupecký, and K. Mikula, "Geometrical Image Segmentation by the Allen-Cahn Equation," *Appl. Numer. Math.*, vol. 51, pp. 187–205, 2004.
- [18] G. Doğan, P. Morin, and R. H. Nochetto, "A Variational Shape Optimization Approach for Image Segmentation with a Mumford-Shah Functional," *SIAM J. Sci. Comput.*, vol. 30, no. 6, pp. 3028–3049, 2008.
- [19] G. Chung and L. A. Vese, "Image Segmentation Using a Multilayer Level-Set Approach," *Comput. Visual. Sci.*, vol. 12, no. 6, pp. 267–285, 2009.
- [20] J. Mille, "Narrow Band Region-Based Active Contours and Surfaces for 2D and 3D Segmentation," *Comput. Vis. Image Und.*, vol. 113, no. 9, pp. 946–965, 2009.
- [21] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour Detection and Hierarchical Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, 2011.
- [22] A. Chambolle, D. Cremers, and T. Pock, "A Convex Approach to Minimal Partitions," *SIAM J. Imaging Sci.*, vol. 5, no. 4, pp. 1113–1158, 2012.
- [23] R. Fergus, P. Perona, and A. Zisserman, "Object Class Recognition by Unsupervised Scale-Invariant Learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. 264–271.
- [24] J. C. Bezdek, J. Keller, R. Krisnapuram, and N. R. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, ser. The Handbooks of Fuzzy Sets Series. Springer, 2005.
- [25] M. S. Nixon and A. S. Aguado, *Feature Extraction and Image Processing*, 1st ed. Oxford, Auckland, Boston, Johannesburg, Melbourne, New Delhi: Newnes, 2002.
- [26] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Comput. Surv.*, vol. 41, no. 3, 2009.
- [27] R. Kimmel and A. M. Bruckstein, "Regularized Laplacian Zero Crossings as Optimal Edge Integrators," *Int. J. Comput. Vision*, vol. 53, no. 3, pp. 225–243, 2003.
- [28] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, "An Algorithm for Minimizing the Mumford-Shah Functional," in *Proceedings of the 12th IEEE International Conference on Computer Vision (ICCV 2009)*, 2009, pp. 1133–1140.
- [29] H. Schaeffer and L. Vese, "Active Contours with Free Endpoints," *J. Math. Imaging Vis.*, vol. 49, no. 1, pp. 20–36, 2014.
- [30] H. Benninghoff and H. Garcke, "Efficient Image Segmentation and Restoration Using Parametric Curve Evolution With Junctions and Topology Changes," *SIAM J. Imaging Sci.*, vol. 7, no. 3, pp. 1451–1483, 2014.
- [31] M. Balažovjeh, K. Mikula, M. Petrášová, and J. Urbán, "Lagrangean Method with Topological Changes for Numerical Modelling of Forest Fire Propagation," in *Proceedings of ALGORITMY 2012, 19th Conference on Scientific Computing*, Vysoké Tatry, Podbansk'v, Slovakia, 2012, pp. 42–52.
- [32] K. Mikula and J. Urbán, "New Fast and Stable Lagrangean Method for Image Segmentation," in *Proceedings of the 5th International Congress on Image and Signal Processing (CISP 2012)*, Chongqing, China, 2012, pp. 834–842.
- [33] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing*. New York: Springer, 2006.
- [34] J. W. Barrett, H. Garcke, and R. Nürnberg, "On the Variational Approximation of Combined Second and Fourth Order Geometric Evolution Equations," *SIAM J. Sci. Comput.*, vol. 29, no. 3, pp. 1006–1041, 2007.
- [35] —, "A Parametric Finite Element Method for Fourth Order Geometric Evolution Equations," *J. Comput. Phys.*, vol. 222, no. 1, pp. 441–467, 2007.
- [36] T. Davis, "Algorithm 832: UMFPACK, an Unsymmetric-Pattern Multifrontal Method," *ACM Trans. Math. Software*, vol. 30, no. 2, pp. 196–199, 2004.
- [37] S. Osher and J. A. Sethian, "Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.
- [38] G. Sapiro, *Geometric Partial Differential Equations and Image Analysis*. New York: Cambridge University Press, 2006.



**Heike Benninghoff** is a research associate at German Aerospace Center (DLR) in Oberpfaffenhofen, Germany, working since 2010 in the field of autonomous optical navigation and control. She received the diploma degree in mathematics from the University of Regensburg in 2010, where she is currently a PhD candidate in applied mathematics. Her research interests comprise visual navigation, image processing and numerical simulation.



**Harald Garcke** received the Ph.D. degree in applied mathematics from Bonn University, Germany, in 1993. He subsequently went on postdoctoral positions at Sussex University (UK) and Bonn where he finished his habilitation thesis in 2000. He became a full professor at the University of Regensburg in 2002. His research interests are nonlinear partial differential equations, geometric evolution equations, numerical analysis, optimization and image analysis.